

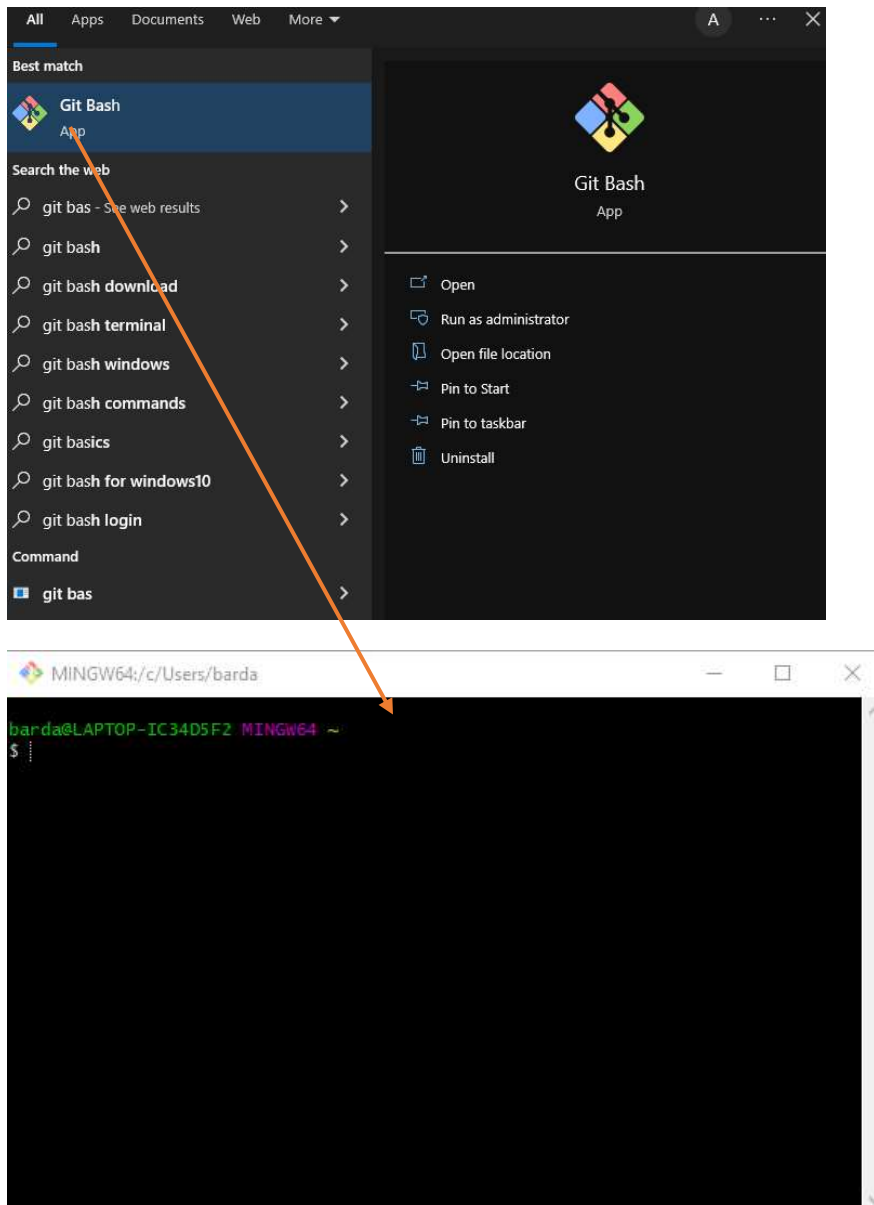
עבודה עצמית 0 – שימוש ב-git ו-GitHub

על מנת לבצע תרגול זה נדרש לבצע את השלבים הבאים לפני (פרטים במצגת של תרגול R01):

1. לוודא שמותקנת תוכנת git על המחשב שלכם.
2. לפתוח משתמש ב-GitHub.
3. לעקוב אחרי ההוראות ליצירת והכנסת צמד מפתחות SSH למחשב האישי שלכם ול-GitHub.
4. לחצתם accept על המטלה HW_0 (קישור במצגת), ונוצר לכם repository ב-GitHub בשם:
HW_0_<your GH user name>

צעד 0:

- נפתח את git bash (מגיעה כחלק מהתקנת git). ייפתח חלון עם command line:



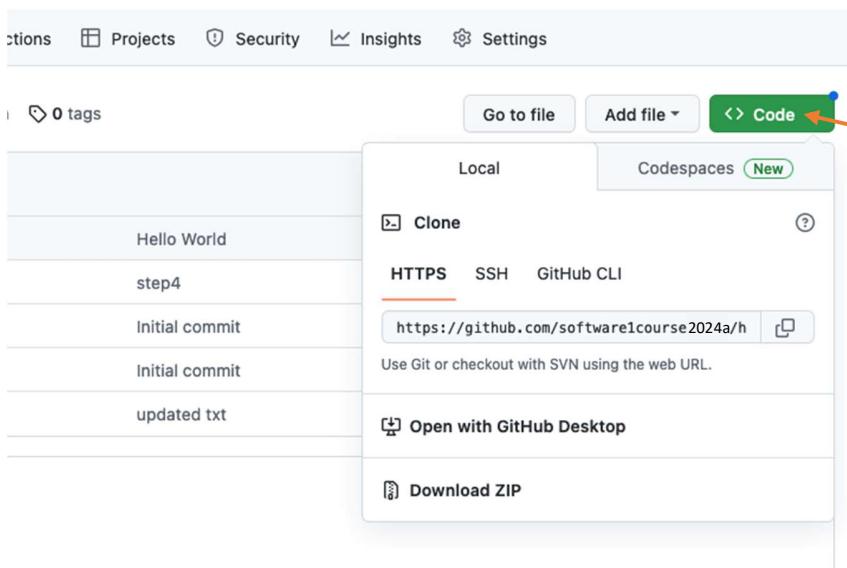
- נשנה את המיקום להיות התיקיה בה נרצה לעבוד על המטלה באמצעות הפקודה `cd <path_to_directory>` ה-path הזה יהיה ה-current working directory (cwd), למשל, בדוגמא שלנו הוא `.C:/Users/barda/Desktop/software1/b`.

```

MINGW64; c:/Users/barda/Desktop/software1/b
bar da@LAPTOP-IC34D5F2 MINGW64 ~
$ cd C:/Users/barda/Desktop/software1/b
bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b
$ |
  
```

צעד 1: יצירת repository מקומי מ-remote repository

ה-repo ב-GitHub הוא ה-"remote repository" (מעתה נקרא לו רק "remote"). נרצה לייצר עותק מקומי על המחשב האישי שלנו "local repository" שמקושר אליו. את העותק המקומי נערוך ונבדוק, ואז נעדכן את ה-repo המרוחק (ה-remote). את העותק המקומי ניצור באמצעות שימוש בפקודה `git clone`. תחילה, נעתיק מה-GitHub את הקישור עבור פעולת ה-clone. ב-repo נלחץ על הכפתור הירוק שרשום עליו Code ואז נעתיק את הכתובת. שימו לב הקישור שאתם מעתיקים יופיע בפורמט: `https://github.com/software1course<current_semester>/hw_<number>`



לחצו כאן כדי לקבל את

ניתן להעתיק את הכתובת ישירות או ללחוץ על הכפתור הזה

לאחר מכן, ב git bash, נריץ את הפקודה:

`git clone <repo>`

כשבמקום <repo> עליכם להזין את הכתובת שהעתקתם. לדוגמא:

```
MINGW64:/c/Users/barda/Desktop/software1/b
barda@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b
$ git clone https://github.com/software1course2024a /hw_0-GH_user.git
Cloning into 'hw_0-GH_user'...
bash.exe: warning: could not find /tmp, please create!
bash.exe: warning: could not find /tmp, please create!
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 13 (delta 2), reused 12 (delta 2), pack-reused 0
Unpacking objects: 100% (13/13), done.

barda@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b
$ |
```

אם הפקודה בוצעה בהצלחה, תיווצר תיקייה ב-cwd שקבענו בצעד 0.

נוודא באמצעות הפיכת התיקייה שנוצרה ל-cwd:

```
barda@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b
$ cd hw_0-GH_user

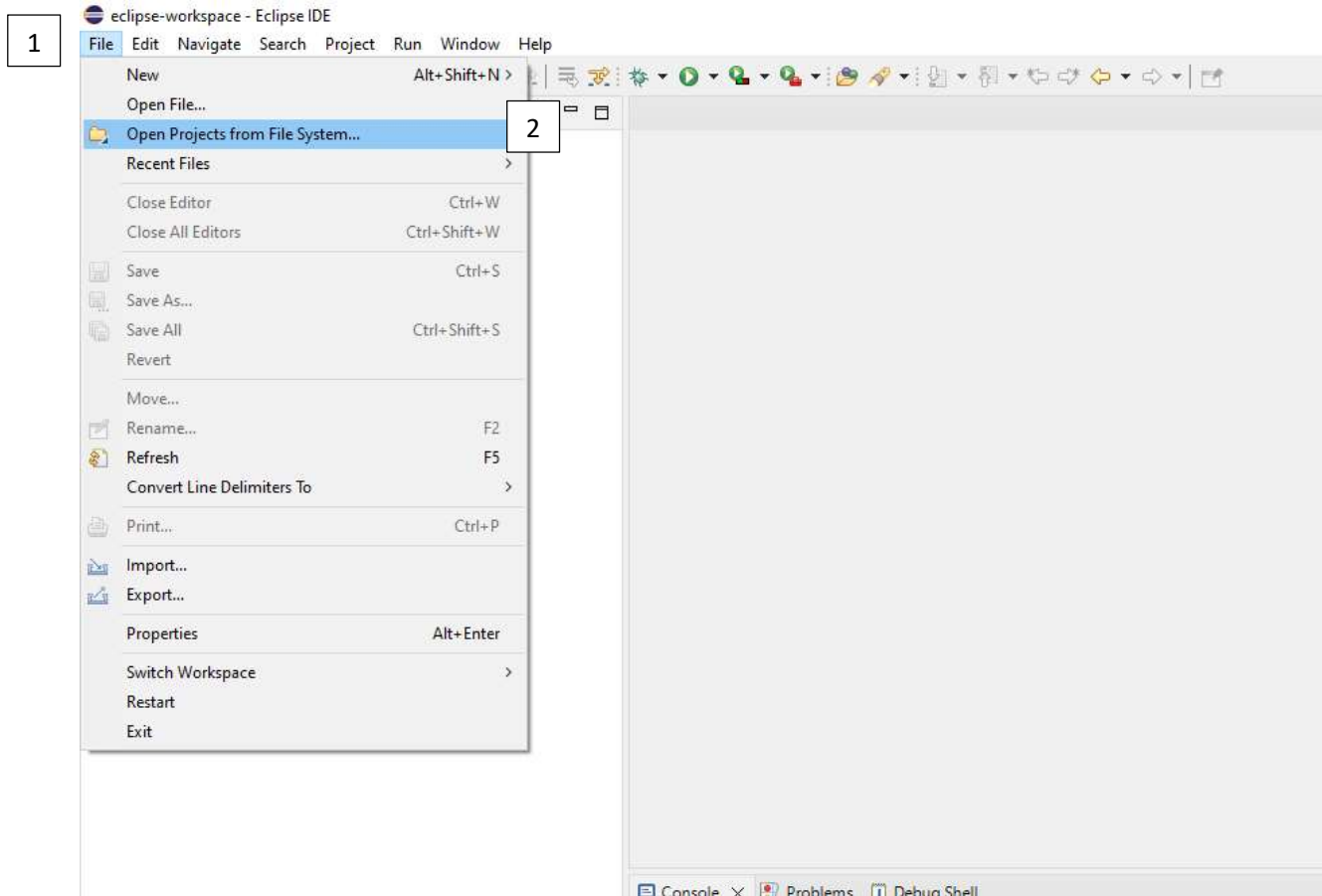
barda@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ |
```

Git מזהה שהתיקייה הנוכחית היא local repository שמקושרת ל-remote. ניתן לראות את זה באמצעות הכיתוב התכלת שמראה את ה-branch שה-local repository נמצא עליו (כפי שראיתם בהרצאה, לאותו הפרוייקט יכולות להיות כמה הסתעפויות, branches. הענף הראשי נקרא main).

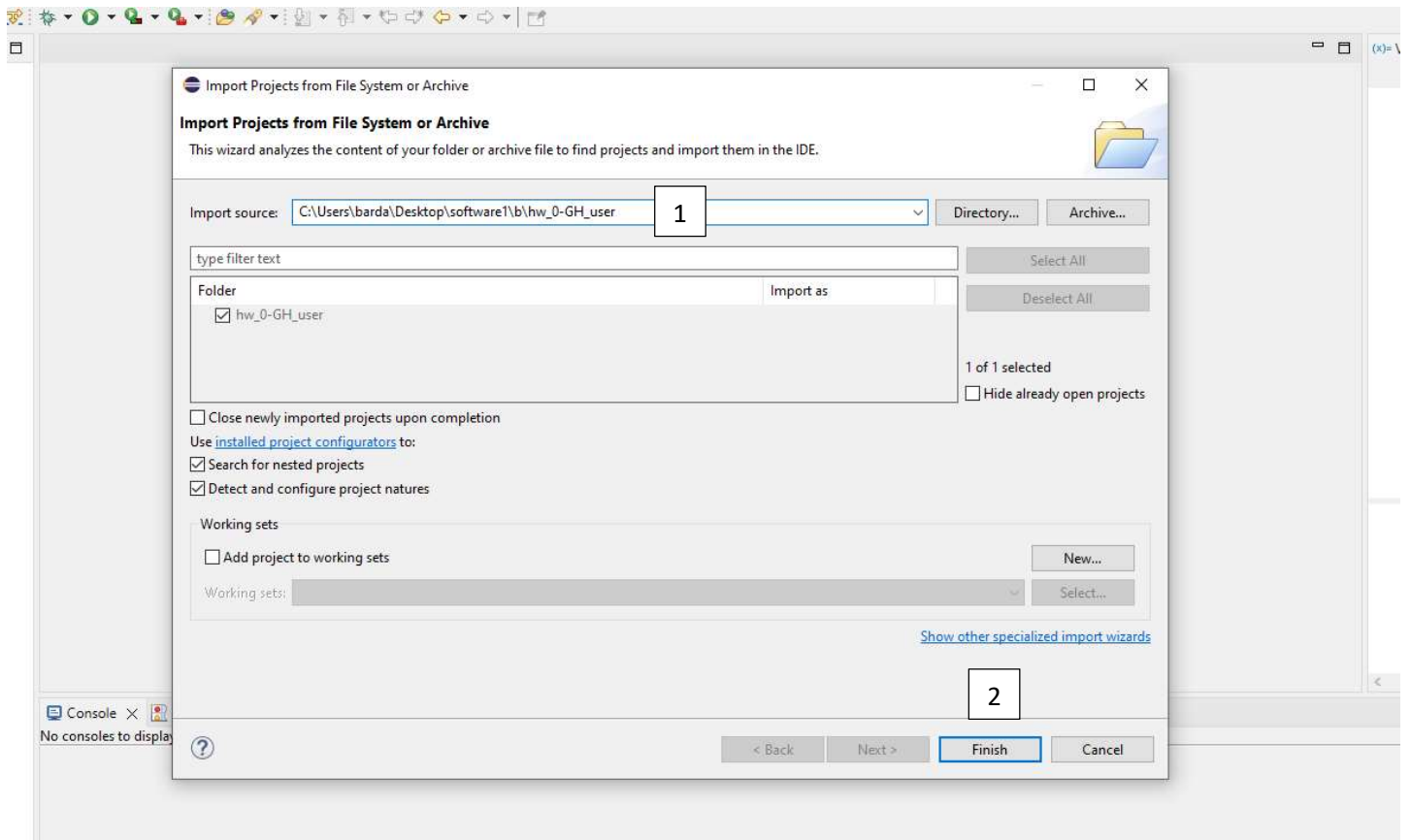
צעד 2: פתיחה של הפרויקט ב-IDE

מדריך זה ראה עבודה ב Eclipse. מי שבוחר להשתמש ב-IDE אחר, באחריותו לבצע את הפעולות המקבילות בסביבת הפיתוח שלו.

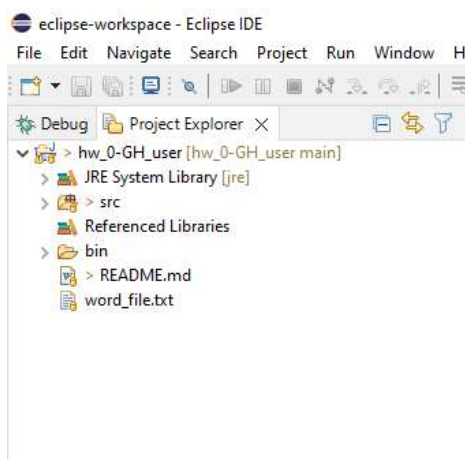
נלחץ על `file > Open Projects From File system ...`



בחלון שנפתח נכניס את ה-path לתיקייה שנוצרה ב-clone ונלחץ finish.



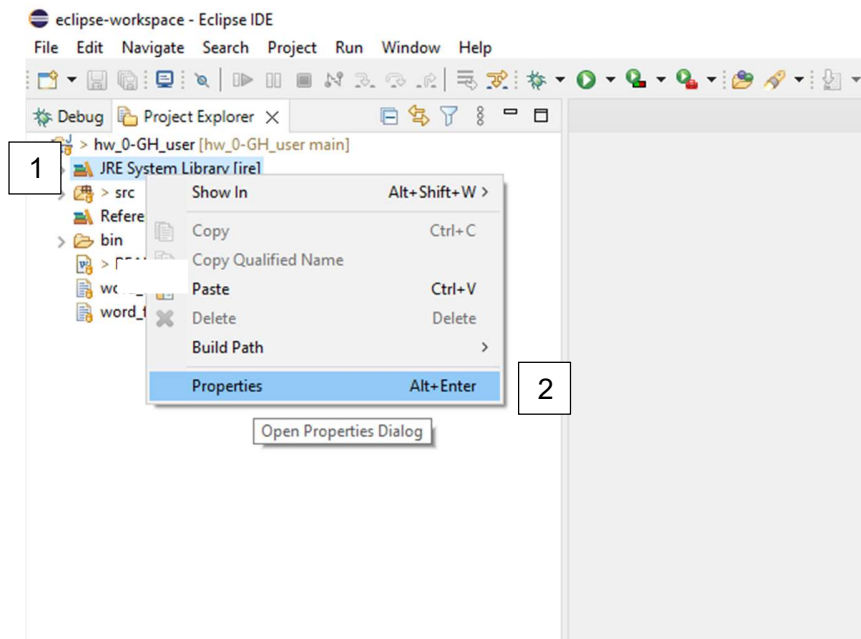
כעת ה Eclipse שלנו מכיר את הפרוייקט.



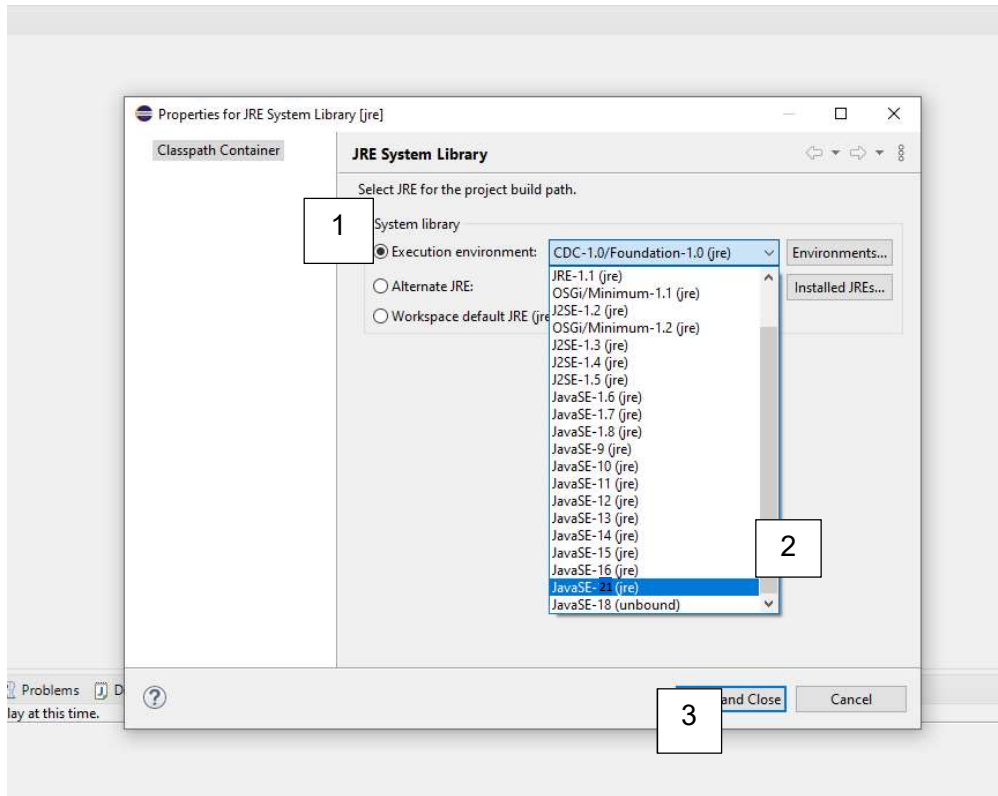
צעד 3: מעבר לסביבת Java21

גרסת ה-Java שנעבוד בה בקורס היא Java21, ויש לוודא שסביבת הפיתוח שלכם מוגדרת להשתמש בה.

ב-Eclipse, נמצא את הפרויקט בסייר הקבצים, ונלחץ קליק ימני על JRE System Library, ונלחץ על .properties



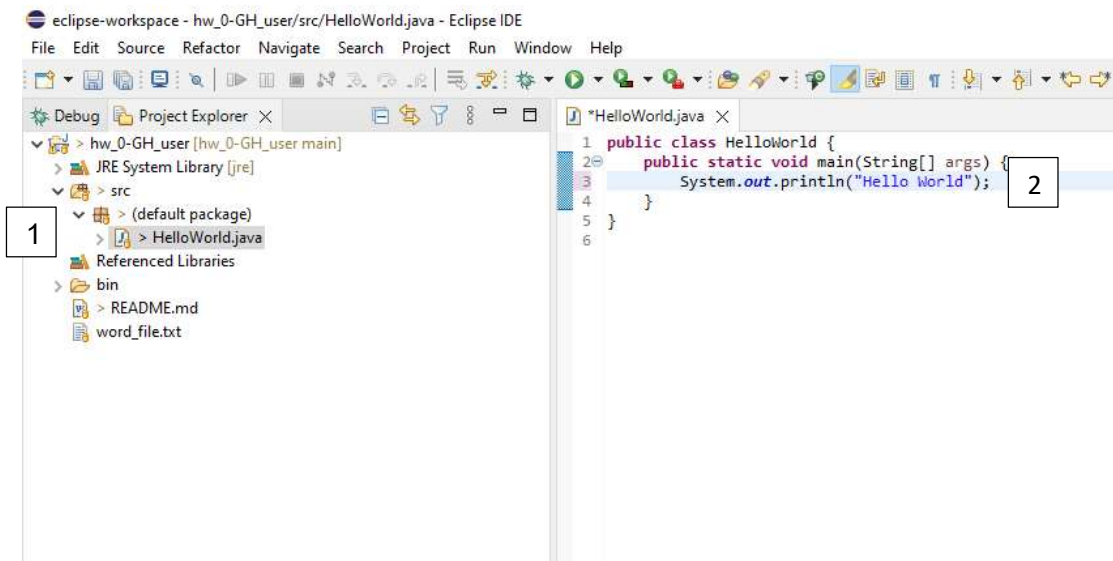
בחלון שנפתח נסמן את Execution Environment, ונבחר את JavaSE-21, ונלחץ Apply and Close.



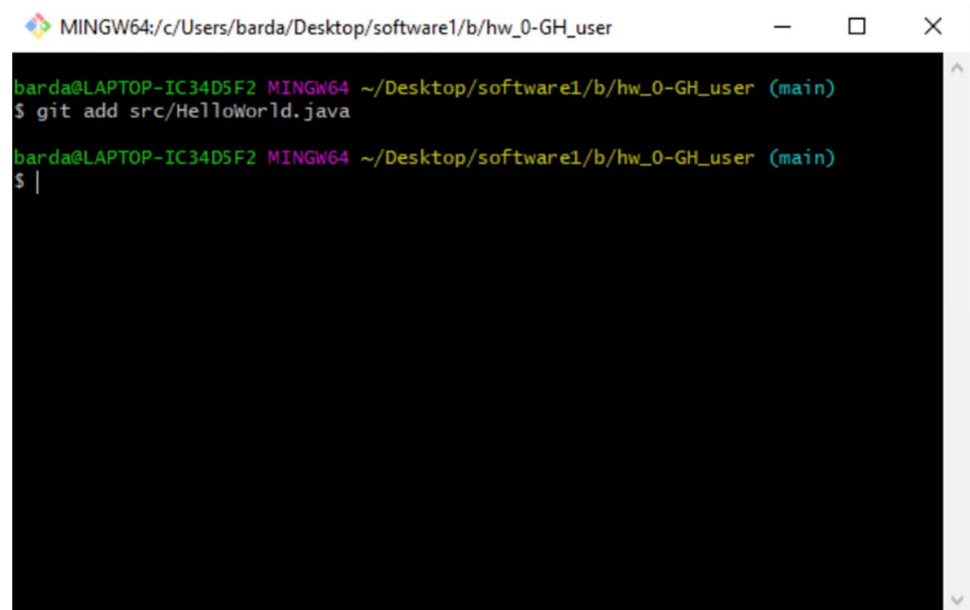
צעד 4: עריכת קבצים קיימים ועדכון ב-remote.

נפתח באמצעות סייר הקבצים את הקובץ HelloWorld.java בתיקיית src.

נערוך את הקוד בכך שנכתוב Hello World בתוך פקודת ההדפסה (println).



ראשית נשתמש בפקודת `git add` עם ה-`path היחסי` (יחסית ל-`cwd`) של הקובץ ששונה:

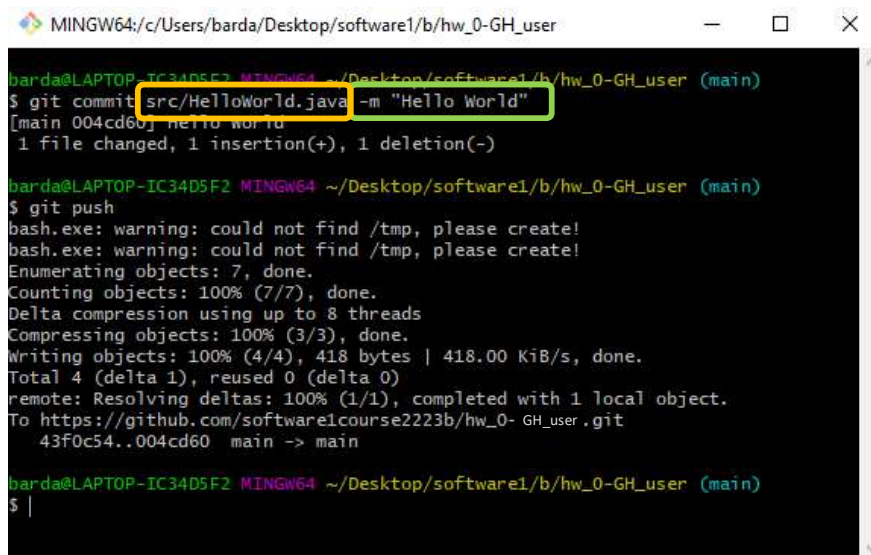


```
MINGW64:/c/Users/barda/Desktop/software1/b/hw_0-GH_user
bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ git add src/HelloWorld.java

bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ |
```

כעת ב-`git bash` נשתמש ב-`git commit` על מנת להכניס את השינוי ל-`local repo` שלנו. יש לשים לב שה-`path` שנותנים הוא יחסי ל-`cwd`, ואנחנו מוסיפים הערה עם הדגל `-m` (ב-`git` חובה להוסיף הסבר לכל `commit`)

לאחר מכן נשתמש ב-`git push` על מנת לדחוף את ה-`commit` ל-`remote`.



```
MINGW64:/c/Users/barda/Desktop/software1/b/hw_0-GH_user
bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ git commit src/HelloWorld.java -m "Hello World"
[main 004cd60] Hello World
1 file changed, 1 insertion(+), 1 deletion(-)

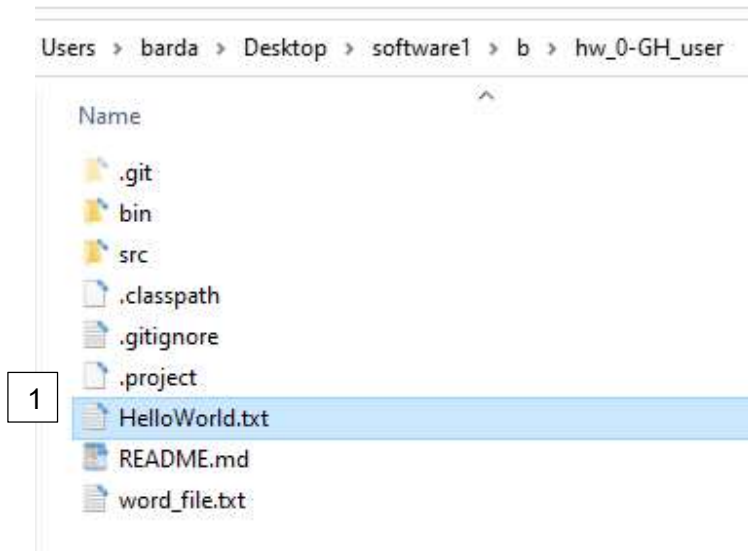
bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ git push
bash.exe: warning: could not find /tmp, please create!
bash.exe: warning: could not find /tmp, please create!
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 418 bytes | 418.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/software1course2223b/hw_0- GH_user .git
 43f0c54..004cd60 main -> main

bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ |
```

במידה וכל הפקודות בוצעו בהצלחה, הקובץ `HelloWorld.java` ב-`remote` התעדכן כעת (תבדקו!).

צעד 5: הוספת קבצים חדשים ל-remote

לצורך ביצוע הצעד הזה ניצור קובץ בשם HelloWorld.txt בתיקיית המטלה.



על מנת להוסיף את הקובץ לבקרה של git, משתמש בפקודת git add.

ואז כמו בצעד 3, נשתמש ב-git commit על מנת להכניס אותו ל-local repository ואז ב-git push כדי להכניס אותו ל-remote.

```
MINGW64:/c:/Users/barda/Desktop/software1/b/hw_0-GH_user
bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ git add HelloWorld.txt

bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ git commit HelloWorld.txt -m "step4"
[main 98a2df7] step4
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 HelloWorld.txt

bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ git push
bash.exe: warning: could not find /tmp, please create!
bash.exe: warning: could not find /tmp, please create!
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 264 bytes | 264.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/software1course2024a/hw_0-GH_user.git
 adffdbd..98a2df7 main -> main

bar da@LAPTOP-IC34D5F2 MINGW64 ~/Desktop/software1/b/hw_0-GH_user (main)
$ |
```

במידה וכל הפקודות בוצעו בהצלחה, הקובץ HelloWorld.txt נוסף ל-remote (בדקו את זה!).